

COSC 201 – Assignment #2

Fall 2016

Objective: Create a small expression evaluator that handles basic integer arithmetic and variables.

Using the example calculator code from your book (see section 11.2 in your book) you will create a calculator that handles a variety of operations on integers, including **variables with integer type**. You should handle the following operations, noting that certain operations can have a different meaning when used in different settings:

Operation	Meaning
+	Addition
-	Subtraction OR negation
*	Multiplication
/	Integer division (i.e. 14 / 3 is equal to 4)
%	Modulo (i.e. 14 % 3 is equal to 2)
**	Power (i.e. 3* *5 is equal to 243)
=	Assignment

You also will need to handle variables. To make things simpler, you should always assume that a valid variable name is a single alphabet character (a-z,A-Z) and that variable should always be initialized before use. For example:

```
> x = 5
x set to 5
> x + 4
9
```

is a good segment of code, while:

```
> x + 4
x is undefined.
```

should be handled.

You should also handle **parentheses** and **order of operations**. Aside from these characters, operators and numbers, you should assume all other characters are invalid and they should produce an error if they exist in the input line (see Testing).

Code Requirements:

You should provide the Java class to handle input from my driver (see Testing). In order to facilitate this, you must follow these specific requirements (remember, capitalization counts):

```
Class name: StackCalculator
Class constructor signature: public StackCalculator()
Primary Method signature: public void processInput(String s)
```

You will be able to test your code with my driver before needing to submit your class. If your code does not work with my driver, check your signatures with the above requirements first. Aside from the above code requirements, I have no specific implementation requirements other than your code should do the heavy lifting. No short cuts.

